

## Case-Based Reasoning Decision Making in Ambient Assisted Living

Davide Carneiro<sup>1</sup>, Paulo Novais<sup>1</sup>, Ricardo Costa<sup>2</sup>, José Neves<sup>1</sup>

<sup>1</sup>DI-CCTC, Universidade do Minho, Braga, Portugal

davide.carneiro@gmail.com, {pjon, jneves}@di.uminho.pt

<sup>2</sup>College of Management and Technology - Polytechnic of Porto, Felgueiras, Portugal  
rfc@estgf.ipp.pt

**Abstract.** Quality on the welfare services in caring, and the trend to minimize the economical and social-political costs that come with such practice, due to the population aging, are paramount nowadays, i.e., health care reform has become the leading policy issue in all latitudes. Indeed, the major thrust of all this research is the perception that escalating costs make the current structure and financing of health care unsustainable. The issue of sustainability is, therefore, the main subject of this paper. As a result, and in order to accomplish this goal, we decided to look to the problem from an user perspective, i.e., the system not only will provide different services, but will be also able to trace the ones more frequently used and to learn about the context in which they happen. As a result, we will have a system that will act and learn according to the preferences and habits of its users, and, simultaneously, will adapt to the environment with the objective of reducing the cost of its practices.

**Keywords:** Ambient Assisted Living, e-Health, Case Based Reasoning, Machine Learning, Simulation

### 1 Introduction

With the population-ageing phenomenon with which we are faced, healthcare and social security systems are on the verge of collapse [1]. The consequences are the exponentially rising of costs, the dropping of the quality of service and the lack or inexistence of specialized care, being the most harmed people with chronic diseases or physical impairments. There is no doubt that changes are needed in this sector and the answer may pass all the way through the introduction of a new paradigm for the healthcare provision act, and the access to new technologies, that will enforce new methodologies for problem solving, namely those based on the concept of agent, with the primary objective of empowering the user or patient, so that he/she can expect to have a normal life.

Under this new paradigm, the treatment will no more be institution centred but, instead, will shift to the user, resulting in a better and cheaper service for both the user and the provider. Although at a first glance this may seem a contradiction, the lower of costs will be possible all through the use of new, economical devices and

technologies, which will allow the monitoring and assessment of the users vital data in real time. This data, once processed, may be used to trigger alarms, recommendations or even actions to assist the ones in need [2].

Following this new approach to problem solving, a growing number of IT healthcare projects, which try to answer to this challenge, are coming up [3, 4, 10, 11]. Our goal is to take these systems a step ahead, empowering them with proactiveness and learning mechanisms, so that they may be able to adapt to its users profiles and provide assistance on a practical base in their day-by-day life.

## 2 The VirtualECare Project

The VirtualECare project [5, 9] aims at a Multi-agent based architecture in support of computational systems that seek to monitor and provide personalized health care services to the persons in need (i.e., its users or patients). These systems will be used to interconnect healthcare institutions, training facilities and leisure centres, shops and patients relatives, on a common network, i.e., the VirtualECare architecture stands for a distributed one with its different nodes answering for a different role, either in terms of a call centre, a group decision support system or a monitoring device, just to name a few.

Indeed, while the information technology is developing and has its impacts on healthcare provision, the competition among service providers is increasingly fierce, i.e., in order to keep an outstanding quality of care and industrial advantage, prominent technology and perfect service are needed to gain patient support and cooperation partners.

On the other hand, while making use of this technology, we are able to integrate products and technology of business partners to provide core business, including system integrations and technical capital construction solutions, i.e., based on an operational strategy of *patient-oriented, service-focused, quality-first, and technology-depended*, it is possible to provide patients with an excellent sustaining added value and one-stop solution and service that can enhance their quality-of-living.

## 3 The Environment

The environment on which this work fits stands for a simulation one, developed for the VirtualECare project [6]. It reproduces all the fundamentals at a home setting, ranging from the owner himself/herself, with their proper behaviour or preferences, to the existing devices, i.e., the simulation comes across several components and is fully configurable [7].

Indeed, the simulation process starts in the configuration of the living spaces. Then it goes throughout the configuration of the user actions, according to three distinct modes: *Full Random*, *Bounded Random* and *Planned*. Under the former mode, the actions are generated in random spaces, with random lengths and at random instants. As this may lead to nonsense actions one has to look to the *Bounded Random* mode, under which we may decide on the actions that may take place on a particular space.

At last, if one wants to plan a user routine, there is the third mode, under which we decide on the user actions, on which space, in which moment and for how long. The last steps are to configure the user vital signs and the external environment, by means of a simulated weather station.

In addition to the simulation environment, a monitoring system had to be considered. It is composed of three main modules: a database, a monitoring and management interface and 1-Wire sensors to measure environmental values like temperature or luminosity. Along with the database and the interface (Figure 1), the system allows for a complete management and monitorization of the home space, recurring to alarms, warnings and common management services.

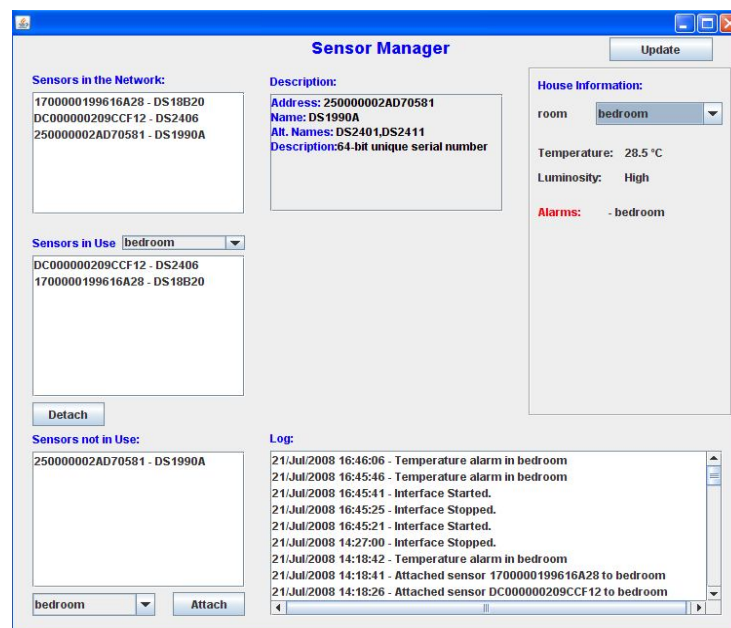


Fig. 1: The Sensor Manager Interface

## 4 Service Selection

In an Ambient Assisted Living environment there is a group of services that are available to the user. In existing projects, the most common is that the user conscientiously engages on an interaction with the services in order to use them. What we propose in this work is that the services may be chosen by the system, without the need for user intervention.

In order to achieve this goal one has to enrich each service with a strong description, i.e., it is not hard, for example, to decide between turning on the light and pulling up the window blind when the luminosity is low inside the house. However,

this only happens once we have our own description of the features of each action. We know that the lights are more effective but are also more energy consuming.

#### 4.1 Service Description

It is vital to identify which are the key parameters that we must take into account when we select a service in our home, in order to do the same for each service in our system. To accomplish this goal the service description is split according to its functionalities, i.e., the functional and the non-functional ones. The former ones describe the service in itself as well as the actions that will be performed when the service is requested. The non-functional ones describe constraints or principles that must be met in order for the service to be available.

In order to get these descriptions it was considered an appropriate set of sentences to describe the services available on a familiar home setting, namely:

- *Alarms* – These sentences identify an event in one space. It may be a fire alarm, a temperature alarm, an intrusion alarm or even having the user pressing the panic button;
- *Actions* – The sentences under this group identify the actions that a service may offer, namely a call to the firefighters or to a patient relative, turning on or off the heat or moving a window blind.
- *Descriptions* – The sentences under this group give a meaning to the statements that come next to them. For example, the sentence **uses** denotes that the service requires exclusivity over a given resource while the sentence **needs** denotes that the service has to use a given resource.

The services are organized in three categories according to its focusing area, being given in terms of security, health or relieve ones. This classification sets the line to be followed in order to get a solution to a problem that we run into while developing this work: the problem of cycling. As an example, let us consider a fire alarm, which leads the system to activate the sprinklers, which, in turn, conduct the system to identify the sprinkler as being the source of the flood alarm. The sprinkler is turned off, which in turn activates the fire alarm, i.e., the process will continue forever.

The solution to these problems includes a way of weighting the substance of the services (its categories), a way of comparing the services inside the same category and the previously mentioned description. Therefore, and according to the example referred to above, the sprinkler and flood alarms are in the same category. However, considering that a fire alarm is more critical than a flood alarm, since it may cause more damage, a service against fire has precedence on a service against water. At last, using the description, the services indicate which resources to lock and which ones are to be enforced.

In this specific example, the fire alarm would request the lock of power and of the sprinkler, and it will succeed, even if they were locked by another service, since the fire alarm is the service with higher priority. Therefore, the sprinkler will be activated. Once the flood alarm is triggered, it would try to gain the control of the sprinkler, but

as it was locked by a service with higher priority, the action will fail and the service would be blocked by the system.

## 4.2 Learning How to Choose Services

In the example mentioned above, the task of service selection is not a hard one, since it only involves basic assessments. However, at an home environment there are judgments that may involve more variables. For example, in a luminosity drop, there are several factors that may influence one decision, namely the power consumption of each service, its effectiveness, the electricity bill, among others. In our everyday life we make these decisions without even noticing that we weight all these parameters. However, a computer system needs to justify all its choices and must have all this data into consideration. More problematic, however, is that we may have the best optimization procedures for a given user, according to its preferences or needs, and the same modus operandi, to another user, may be far from optimum. We maintain that such systems should not be static, but must be able to adapt to its users profiles. Our system was set in order to have in it the potential to learn the preferences and habits of its users and act accordingly. To achieve this goal, we will use the Case Based Reasoning (CBR) approach to problem solving [8].

All the processes are triggered by an alarm (Figure 2). Every time an event happens inside the house, the system creates a new case. This case may be seen as a snapshot of the state of the house environment in that particular moment. A complete case, as it is stored in the knowledge base, holds information about some items, namely:

- *Room* – the space in which the alarm occurred;
- *Alarm* – the alarm which the case refers to;
- *Service* – the service used by the user to solve the alarm;
- *Value* – a value between -1 and 1 which denotes the usefulness of this case when chosen by the system;
- *Time* – the instant of time at which the alarm occurred; and
- *Sensors* – the information about all the sensors in the room and the weather station sensors.

As an example, let us look at the following case:

```
room:- bedroom1
alarm:- lowtemp
service:- ac:tempup
value:- 0
time:- 1231429260828
sensors:- 1700000199616A28:15.54,
8E00000199765928:16.01, 6600000020AD1E12:high, ...
```

When an alarm occurs, the system searches in the knowledge base for the case which is more similar to the one just created (the *Retrieve* phase). This is a very

important step and relies on the similarity function for deciding which cases are similar to the new one. In this case, the function is the sum of differences for the values of the sensors. This means that if the sum of the difference of values of sensors of the same type is small, the case is similar to the new case. If, however, some values for similar sensors show some discrepancy, it means that the cases are not that similar. Other important factor in the similarity function is the room at which the cases occur, i.e., cases that took place in the same room will be more similar than cases that did not. This however is not as significant as the values of sensors. After selecting the closest case, two things may occur: the case is so close that the case action may be subscribed by the system, or the user interaction is requested, which means that we are faced with a new case.

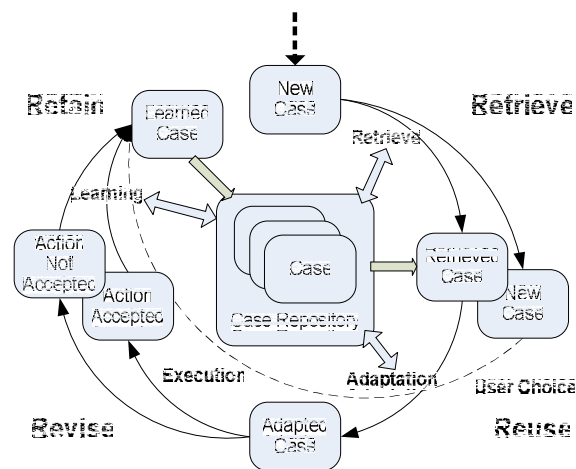


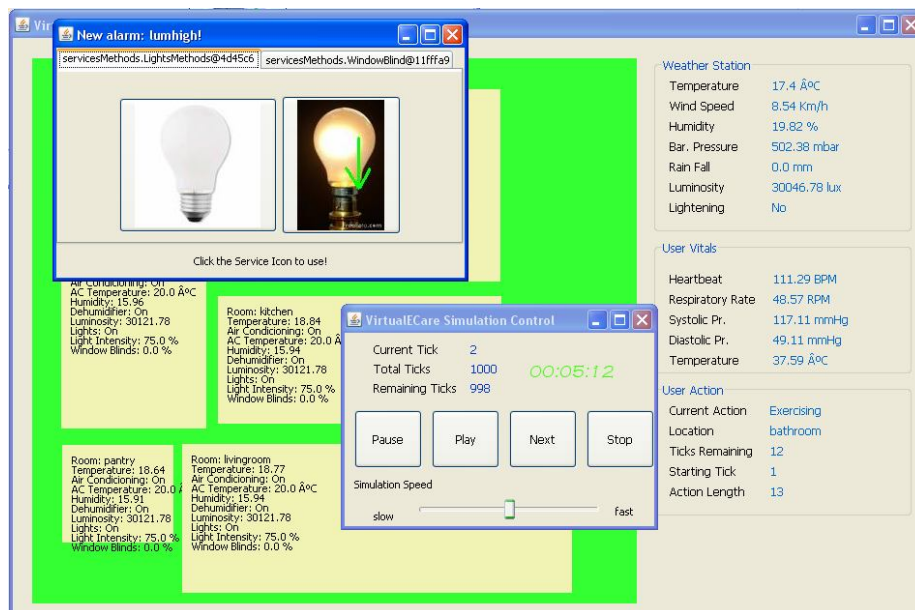
Fig. 2: The Case Based Reasoning Model.

Under the former situation the case may either be used, if its similarity is in line with the expected, or it may be adapted (the *Reuse* phase). Adapting cases is a relatively common practice, and arises when there are cases that are similar on its sensors values but came about in different locations. The adaptation takes place by creating a new case with all the parameters of the selected case but referring to the place where the alarm occurred. The new case is then added to the knowledge base. On the other hand, whenever an action is triggered by the system, the behaviour of the user is evaluated.

However, if during a agreed period of time the user does not react to the action that was triggered, the significance of the case raises. This tells the system that the case is acceptable and will be preferred in future times. Conversely, if the user triggers an action, that behaviour has to be studied. Here, the CBR process enters into the *Revise* phase, being assessed the outcome of the enforcement of the selected case, which in turn may be used to improve the overall performance of the system itself. Furthermore, if the user simply cancels an order that was issued by the system, that means that the selected case was the proper one, i.e., the action performed was the expected, but was not triggered at the right time. As an example, let us consider that the alarm temperature was activated and the system turned on the air conditioning,

and that the user, going round, turned it off, i.e., the user may want the air conditioning to be turned on only and only if when the home temperature grows a bit. As a consequence, the limit of the temperature alarm for that room rises, but the case rate remains the same. Continuing with the same example, if the user besides cancelling the system action, considers another that aims at the same objective, this means that the time at which the alarm occurred was the expected, but the action triggered may not had been the right one. A good example of this is having the system turning on the lights because the low luminosity alarm was activated and, after that, the user turning the lights off, and proceeds moving the window blind up. The effects of this happening are that the value of the existing case is decreased a little bit and a new case is added, describing the state of the house and the action taken. Every time the process reaches its end (the *Retain* phase), the knowledge base is enriched with the information being learned.

When there are no cases that are similar enough and the user interaction is requested (Figure 3), the services that are available and suited to solve the alarm problem are selected and shown to the user, so that he/she can select the more appropriate case. Then, the system inserts the information about the action taken by the user into the new case, and the case is added to the knowledge base.



**Fig. 3:** A Simulation Snapshot

It was shown that using CBR it is possible to build a system that will fit into the user profile, and will keep doing it forever. It has the advantage of adjusting its behaviour even without a direct interaction from the user, but throughout an analysis of its behaviour, that is according its habits or preferences.

## 5 Conclusions and Future Work

As a result of this work it was developed a prototype of a CBR system that exploits the users profiles in terms of its own machine learning capabilities to provide a service or services in an Ambient Assisted Living milieu, i.e., it is dynamic in that it will evolve as a consequence of its interaction with the user and the environment, being effective even when the user changes its habits or routines.

On the other hand, the development of hybrid representations for Machine Learning systems have resulted in several interesting approaches which combine Rule-Induction (RI) methods with CBR techniques to engender performance improvements over more traditional one representation models. CBR is used in learning and problem-solving systems to solve new problems by recalling and reusing specific knowledge obtained from past experience. RI systems learn general domain-specific knowledge from a set of training data and represent the knowledge as IF-THEN rules. These rules are captured after some autonomous accepted actions derivate of the CBR learning. These rules will be valid until the user cancels one of those actions, in which case CBR learning will take place again.

## References

1. United Nations: World Population Ageing: 1950-2050. UN. (2002)
2. IBM Global Business Services: Healthcare 2015: Win-win or lose-lose?. (2006)
3. Holmlid, S., Björklind, A.: Ambient Intelligence to Go. AmIGo White Paper on mobile intelligent ambience. (2003)
4. Fraile J.A., Bajo J., Abraham A., and Corchado J.M.: TITLE: HoCaMA: Home Care Hybrid Multiagent Architecture. In: Advances in Intelligent Multimedia and Pervasive Computing, Part II - Ambient Intelligence and Ubiquitous Computing, Springer Verlag. (2008)
5. Novais P., Costa R., Carneiro D., Machado J., Lima L., Neves J., Group Support in Collaborative Networks Organizations for Ambient Assisted Living, in Towards Sustainable Society on Ubiquitous Networks, Makoto Oya, Ryuya Uda, Chizuko Yasunobu (eds), Springer-Verlag, Series: IFIP International Federation for Information Processing, ISBN 978-0-387-85690-2, pp 353-362, 2008.
6. Carneiro D., Costa R., Novais P., Neves J., Machado J., Neves J.: Simulating and Monitoring Ambient Assisted Living. In: Proceedings of the ESM 2008 - The 22nd annual European Simulation and Modelling Conference, Le Havre, France, October, ISBN 978-90-77381-44-1, pp. 175-182. (2008)
7. OSGi Alliance: OSGi Service Platform, Rel. 3. IOS Press. (2003)
8. Aamodt A., Plaza E.: Case-based reasoning: Foundational issues, methodological variations, and system approaches. In: AI Communications, 7(1):39-59. IOS Press (1994).
9. Costa R., Carneiro D., Novais P., Lima L., Machado J., Marques A., Neves J.: Ambient Assisted Living. In: Advances in Soft Computing, Vol. 51, Springer-Verlag, pp. 86-94 (2008).
10. Alamán X., Ballesteros F., Bravo J., Fernández D.: Ambient Intelligence at Home: Facts and Future. In: Cepis Upgrade: Ambient Intelligence. Vol. VIII, issue 4. Novatica. (2007)
11. Augusto J., McCullagh P., Ambient Intelligence: Concepts and Applications. Computer Science Inf. Syst. 4(1): 1-27 (2007).